Basic Configuration of the Cisco 12000 Series Internet Router

This chapter describes how to boot and configure the Cisco 12000 Series Internet Router. It discusses the following subjects:

- Cisco IOS Software Images, page 2-1
- Configuring the Router, page 2-3
- Configuration Tasks, page 2-5
- Route Processing, page 2-9
- PCMCIA Flash Memory Tasks, page 2-11
- Upgrading and Backing Up Cisco IOS Software Images and Configuration Files, page 2-13

For information on installing your router or LED indications during the boot process, refer to your installation and configuration guide.

Cisco IOS Software Images

By default, your router ships with a Cisco IOS software image preloaded into the Flash memory card or Flash disk inserted in PCMCIA slot 0 on the Route Processor (RP). Initially, your router is configured to boot from this image. You can change this default and configure the router to boot from a Flash memory card or Flash disk in slot 1 on the RP, or from a TFTP server. If you specify that the router boot from an image on a TFTP server, you must verify that you have an Ethernet connection to the URL of the TFTP server.

In addition, there is a mini-software image preloaded into the single inline memory module (SIMM) or bootflash. This image can be used to boot the router if there is no other valid image available. It provides a limited number of configuration commands that you can use to locate a valid Cisco IOS software image or correct basic configuration problems. This image provides no routing capabilities.

Related issues are discussed in the following sections:

- Locating a Valid Cisco IOS Software Image, page 2-2
- Manually Booting from a Cisco IOS Software Image, page 2-2

Locating a Valid Cisco IOS Software Image

If your router does not find a valid system software image, the system should enter read-only memory (ROM) monitor mode and display the ROM monitor prompt (Rommon>). From this mode, you can use the following commands to locate a valid system image.

Command	Purpose
Rommon 1> **dir bootflash:**	Displays the contents of the single inline memory module (SIMM), also known as the bootflash, on the route processor (RP). Images in the bootflash are mini-software images that are used for basic configuration tasks only.
Rommon 2> **dir slot0:** or Rommon 2> **dir slot1:**	Displays the contents of the Flash memory card (if present) in either PCMCIA slot 0 or slot 1.
Rommon 3> **dir disk0:** or Rommon 3> **dir disk1:**	Displays the contents of the Flash memory disk (if present) in slot 0 or slot 1.

The following examples show the output from these commands:

```
Rommon 1> dir bootflash:
Directory of bootflash:/

    1  -rw-     5043356   Jan 01 2000 00:01:15  gsr-boot-mz.120-8.S
Rommon 2> dir slot0:
Directory of slot0:/

    1  -rw-    13778192   Jan 14 2002 18:24:26  gsr-p-mz.120-19.S.bin
   12  -rw-        3973   Sep 03 2002 23:09:47  ozRun

20578304 bytes total (6762344 bytes free)
```

Manually Booting from a Cisco IOS Software Image

If your router does not find a valid system software image, you will need to boot an image manually by issuing the appropriate ROM monitor mode **boot** command. Once you have located a software image, use one of the following forms of the **boot** command to boot the router:

Command	Purpose
Rommon 3> **boot**	(No argument.) Boots the default image found in the SIMM or bootflash. This image is preloaded into the SIMM at the factory.
Rommon 3> **boot flash**	(Does not specify a particular PCMCIA slot.) Attempts to boot the router using the first file found in the Flash memory card inserted in slot 0 of the RP.
Rommon 3> **boot slot0:** *filename*	Boots the router using the specified file on the Flash memory card in slot 0 of the RP.
Rommon 3> **boot disk0:** *filename*	Boots the router using the specified file on the Flash memory disk in slot 0 of the RP.
Rommon 3> **boot slot1:** *filename*	Boots the router using the specified file on the Flash memory card in slot 1 of the RP.

Command	Purpose
`Rommon 3> boot disk1: filename`	Boots the router using the specified file on the Flash memory disk in slot 1 of the RP.
`Rommon 3> boot bootflash: filename`	Boots the router using the specified file on the SIMM (bootflash) on the RP.
`Rommon 3> boot tftp: filename [host]`	Boots the router using the specified file on a host TFTP server in the network.

If you did not change the contents of the software configuration register, the factory default setting of 0x0102 in the software configuration register causes the system to boot from the Cisco IOS software image on a Flash memory card or Flash disk inserted in PCMCIA slot 0 the next time you boot the router.

Configuring the Router

You can perform a basic configuration for your router by using either of the following methods:

- Setup facility or the **setup** command.
  This method provides an interactive script to guide you through the configuration process. It is described in the "Configuring the Router Using the Setup Command" section on page 2-3.
- Global configuration mode through the Cisco IOS command line user interface.
  This method requires you to enter configuration commands on a line-by-line basis at the console, without being prompted by a configuration script. It is described in the "Using Global Configuration Mode" section on page 2-4.

You can use whichever method suits your operating style and your knowledge of network configuration requirements.

Whether you choose to use the **setup** command facility or the global configuration mode to configure the router to operate in your networking environment, be sure you know the following before starting the configuration procedure:

- Interfaces on the router
- Protocols the router is routing
- Network addressing scheme for the router
- Password scheme for your environment

Note Before you can configure the router, a console terminal should be connected to the console port on the faceplate of the RP. For more information on router installation and boot procedure, refer to your installation and configuration guide.

Configuring the Router Using the Setup Command

One method of configuring the router is to use the **setup** command, also known as the **setup** command facility. During the first startup of an unconfigured router, the system automatically starts the **setup** command facility, which enables you to begin configuring your router. The **setup** command facility presents a structured, interactive script that guides you through the configuration process with prompts for global (system-wide) parameters and interface (line card) parameters.

You can also invoke the **setup** command facility at any time by issuing the **setup** command at the privileged EXEC mode prompt (`Router#`), which invokes the same configuration script that appears automatically during the first startup of an unconfigured router. You can use the **setup** command to alter previously entered configuration information or to enter a new configuration. The advantage of using the **setup** command facility is that the system uses an interactive script to guide you through the configuration process.

If you use the **setup** command facility to alter the router configuration, you must allow the entire **setup** command facility script to run, until you come to the item that you want to change. To accept default settings for items that you do not want to change, press the console keyboard **Return** key. To return to the privileged EXEC prompt without making changes, press **Ctrl-C**. To access help text in the setup command facility, press the question mark key (?) at any prompt.

When you complete your changes, the **setup** command facility displays the configuration command script that was created as a result of the changes you entered during the setup session. It also queries if you want to use this configuration. If you answer Yes, the configuration is saved to NVRAM. If you answer No, the configuration is not saved and the process begins again. There is no default for this prompt; you must answer either Yes or No.

Following is an example of the initial output from the **setup** command facility session:

```
--- System Configuration Dialog ---

Continue with configuration dialog? [yes/no]: Yes

At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.

Basic setup only configures enough connectivity
for management of the system, extended setup will ask you
to configure each interface of the system.
```

Note The only observable difference between the configuration script displayed when the **setup** command facility starts automatically on startup, and the script displayed when you manually enter the **setup** command, is that when you enter the command manually, the script displays any previously entered system configuration defaults within square brackets ([ ]).

Using Global Configuration Mode

If you prefer not to use the interactive script of the setup facility, you can manually configure your router using global configuration mode. Global configuration mode enables you to enter configuration commands line by line from the console terminal.

Before you can configure your router using the global configuration mode, you will need to be familiar with the Cisco IOS software command line interface. If you are unfamiliar with the Cisco IOS command line interface, you should read the “Using the Command Line Interface” chapter in the *Configuration Fundamentals Configuration Guide*. That chapter discusses the different command modes, context-sensitive help, and editing features.

To configure your router using global configuration mode, follow these steps:

	Command	Purpose
Step 1	`Continue with configuration dialog? [yes/no]: no`	Enters user EXEC mode on the router without entering the **setup** command facility. You receive this prompt when you initially boot the router. If you have completed the **setup** command facility, you will already be in user EXEC mode and can continue with Step 2.
Step 2	`Router> enable` `Router#`	Enters privileged EXEC mode on the router. Depending on the system and the software version, you may be prompted for a password. The prompt changes to `Router#` in privileged EXEC mode.

Step 3	`Router# configure terminal`
	`Enter configuration commands, one per line. End with CNTL/Z.`
	`Router(config)#`
		Enters global configuration mode, from which you can enter most of the configuration commands needed to change the system configuration. The prompt changes to `Router(config)#` in global configuration mode. When you are finished entering configuration commands, press **Ctrl-Z** to exit global configuration mode.

Step 4	`Router(config)# interface` *type slot/port* `Router(config-f)#`	Enters interface configuration mode for the specified interface. In interface configuration mode, you can enter commands to change the interface configuration. The prompt changes to `Router(config-f)#` in interface configuration mode. When you are finished entering configuration commands, press **Ctrl-Z** to exit the configuration mode and return to privileged EXEC mode. Use the **exit** command to return to global configuration mode.

Step 5	`Router# copy running-config startup-config`
	or
	`Router# write memory`
		Saves the running configuration changes to NVRAM. If you do not save the running configuration to NVRAM, your configuration settings will be lost the next time you reload the router.

In global configuration mode, you will enter all the necessary commands to configure your router. The remainder of this document describes typical configuration tasks that you may need to perform. To display a list of the configuration commands available to you, enter a question mark (**?**) at the prompt or press the designated **help** key on the terminal keyboard while in configuration mode. For more information concerning Cisco IOS configuration commands, refer to the Cisco IOS Command Reference located at:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/12cgcr/rbkixol.htm

Configuration Tasks

This section details initializing configuration tasks that should be performed on your router. Each task can be performed using either the setup command facility or the global configuration mode.

- Configuring the Router Host Name, page 2-6
- Configuring Passwords on the Router, page 2-6
- Configuring Ethernet Access for Network Management, page 2-7
- Configuring Line Card Interfaces, page 2-8

Configuring the Router Host Name

The default host name for all routers is "router". You can change the host name by using the **setup** command or the **hostname** command.

Command	Purpose
`Router> hostname name`	Assigns a host name to the router.

When the **setup** command is used, you are prompted to enter a host name, as in the following example:

```
Would you like to enter basic management setup? [yes/no]: Yes
Configuring global parameters:
  Enter host name [Router]: Filo
```

The name you assign the router must follow the rules for ARPANET host names. It must start with a letter, end with a letter or digit, and have as interior characters only letters, digits, and hyphens. The name must consist of 63 or fewer characters. For more information, refer to *Requests For Comments (RFC) 1035, Domain Names—Implementation and Specifications*.

Upper- and lowercase characters look the same to many Internet software applications; therefore for ease of use, computer names should appear in all lowercase. For more information, refer to *RFC 1178, Choosing a Name for Your Computer.*

Configuring Passwords on the Router

The commands available at the user EXEC level are a subset of those available at the privileged EXEC level. Many privileged EXEC commands are used to set system parameters, so you should password-protect these commands to prevent their unauthorized use. Following is a subset of the password protection commands, which are accessed via global configuration mode.

Command	Purpose
`Router(config)# enable password password`	Configures authentication to access the privileged EXEC commands. After using this command, when you use the **enable** command to enter privileged EXEC mode, you will be prompted for the enable password.
`Router(config)# enable secret password`	Configures authentication to access the privileged EXEC commands. The **enable secret** offers better security than the **enable password** because the **enable secret** password is stored using a nonreversible cryptographic function.

`Router(config)# line vty 0 4`
`Router(config-line)# login`
`Router(config-line)# password password`
	Configures authentication to access a router via incoming Telnet sessions.

`Router(config)# line console 0`
`Router(config-line)# login`
`Router(config-line)# password password`
	Configures authentication to access the console terminal.

Password protection can also be configured using the **setup** command facility, as in the following example:

```
The enable secret is a password used to protect access to
privileged EXEC and configuration modes. This password, after
entered, becomes encrypted in the configuration.
```

```
Enter enable secret [<Use current secret>]: esecret

The enable password is used when you do not specify an
enable secret password, with some older software versions, and
some boot images.
Enter enable password: epassword

The virtual terminal password is used to protect
access to the router over a network interface.
Enter virtual terminal password: tpassword
```

For maximum security, the **enable secret** and the **enable password** should be different. If you use the same password for both the **enable secret** and **enable password** prompts during the **setup** process, the system accepts it but issues a warning indicating that you should enter a different password.

An **enable secret** can contain from 1 to 25 uppercase and lowercase alphanumeric characters; an **enable password** can contain any number of uppercase and lowercase alphanumeric characters. You cannot use a number as the first character. Spaces, however, are valid password characters. For example, `two words` is a valid password. Leading spaces are ignored, but trailing spaces are recognized.

Make a note of all passwords you set and store that information in a secure location for future reference.

For more detailed information on how to establish password protection or configure privilege levels, refer to the *Security Configuration Guide*, located at:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/12cgcr/secur_c/index.htm

Configuring Ethernet Access for Network Management

You can configure Ethernet connectivity to your router via the route processor (RP) for network management purposes. The RJ-45 and MII receptacles on the faceplate of the GRP and the RJ-45 receptacles on the faceplate of the PRP are IEEE 802.3u-compliant interfaces. You can use one interface or the other, but not both at the same time.

To configure Ethernet connectivity, perform the following procedure. The IP address and subnet mask are examples only. You will need to obtain this information from your network administrator.

	Command	Purpose
Step 1	`Router(config)# interface ethernet 0`	Enters interface configuration mode on the Ethernet interface.
Step 2	`Router(config-if)# ip address 10.10.1.1 255.255.0.0`	Specifies the IP address and subnet mask for the interface.

Ethernet connectivity can also be configured using the **setup** command facility, as in the following example:

```
Configuring interface Ethernet0:
  Is this interface in use?: yes
  Configure IP on this interface?: yes
    IP address for this interface: 10.10.1.1
    Number of bits in subnet field: 8
    Class A network is 10.0.0.0, 8 subnet bits; mask is 255.255.0.0
  Configure CLNS on this interface?: yes
```

Configuring Line Card Interfaces

Because of the wide variety of line cards supported by the Cisco 12000 Series Internet Router, you should refer to the configuration note that is available for each particular card for interface configuration information. This section provides several examples to demonstrate how to configure line cards using both the global configuration mode and the setup facility.

Configuring a POS Line Card

To configure a POS line card, perform the following procedure. The IP address and subnet mask are examples only. You will need to obtain this information from your network administrator.

	Command	Purpose
Step 1	`Router(config)# interface pos 3/0`	Enters interface configuration mode on the POS interface located in slot 3 and port 0.
Step 2	`Router(config-if)# ip address 2.1.1.1 255.0.0.0`	Specifies the IP address and subnet mask for the interface.
Step 3	`Router(config-if)# no shutdown`	Changes the shutdown state to up and enables the interface. If you configure an interface using the command line interface, you must use this command to enable the interface. When you use the **setup** command facility, this is performed automatically.

Additional configuration tasks can be performed for the line card in interface configuration mode. Refer to the installation note provided with the line card for additional information.

Note By default, POS interfaces use the 32-bit cyclic redundancy check (CRC) and high-level data link control (HDLC) as the encapsulation protocol.

The following commands are also useful for configuring POS interfaces:

Command	Purpose
`Router(config-if)# pos scramble-atm`	Enables SONET payload scrambling on a POS interface.
`Router(config-if)# clock source internal`	Controls which clock a POS interface will use to clock its transmitted data.

The following sample excerpt from a **setup** command facility session for a 4-port OC-3c/STM-1c POS line card shows settings for a typical configuration:

```
Configuring interface POS3/0:
  Is this interface in use?: yes
  Configure IP on this interface?: yes
  Configure IP unnumbered on this interface?: no
    IP address for this interface: 2.1.1.1
    Number of bits in subnet field: 0
    Class A network is 2.0.0.0, 0 subnet bits; mask is 255.0.0.0
  Configure CLNS on this interface?: yes
```

If additional configuration is required, use the global configuration mode as described at the beginning of this section.

Configuring an ATM Line Card

To configure an ATM line card, perform the following procedure. The IP address and subnet mask are examples only. You will need to obtain this information from your network administrator.

	Command	Purpose
Step 1	`Router(config)# interface atm 1/0`	Enters interface configuration mode on the ATM interface located in slot 3 and port 0.
Step 2	`Router(config-if)# ip address 1.1.1.2 255.0.0.0`	Specifies the IP address and subnet mask for the interface.
Step 3	`Router(config-if)# no shutdown`	Changes the shutdown state to up and enables the interface. If you configure an interface using the command line interface, you must use this command to enable the interface. When you use the **setup** command facility, this is performed automatically.

Additional configuration tasks can be performed for the line card in interface configuration mode. Refer to the installation note provided with the line card for additional information.

In the following example, an ATM line card is being configured to use IP.

```
Configuring interface ATM1/0:
  Is this interface in use?: yes
  Configure IP on this interface?: yes
    IP address for this interface: 1.1.1.2
    Number of bits in subnet field: 0
    Class A network is 1.0.0.0, 0 subnet bits; mask is 255.0.0.0
```

Note You might have to configure additional parameters for the installed ATM line cards if you want to use all their capabilities. For example, additional steps are required to configure permanent virtual circuits (PVCs).

Route Processing

In Cisco IOS Release 12.0(22)S, the Performance Route Processor (PRP) was introduced to replace the Gigabit Route Processor (GRP). The PRP provides a faster processor, multilayer cache, improved fabric interface for faster communication between PRP and line cards, and larger memory capacity.

The PRP is designed to support all Cisco IOS software features that are supported on the GRP. Certain exceptions apply regarding High Availability. Refer to the *Cisco IOS Release 12.0S Release Notes* for more information regarding supported features.

For more information regarding the PRP, refer to the PRP Feature Module at the following location:

http://wwwicisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120limit/120s/120s22/prp htm

Redundant Route Processor Support

As of Cisco IOS Release 12.0(5)S, the Cisco 12000 Series Internet Router supports the installation of two route processors (RPs). One RP functions as the primary or active processor. The primary RP supports all normal RP operation. The other RP functions as the secondary or standby processor. The secondary RP monitors the primary and will take over normal RP operations if it detects a failure in the primary RP.

Note The redundant RP features are only supported when using two GRPs or two PRPs. There is no support for redundancy when a GRP is used together with a PRP.

Refer to the following document for further information:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios112/ios112p/gsr/gsr_rp.htm

Route Processor Redundancy Plus (RPR+)

An enhancement to RP redundancy called Route Processor Redundancy Plus (RPR+) was introduced in Cisco IOS Releases 12.0(17ST) and 12.0(22)S. With RPR+, the standby RP is fully initialized and configured. This feature allows RPR+ to dramatically shorten the switchover time if the active RP fails or if a manual switchover is performed. Because both the startup configuration and the running configuration are continually synchronized from the active to the standby RP, line cards are not reset during a switchover. The interfaces remain up during this transfer, so neighboring routers do not detect a link flap (that is, the link does not go down and back up).

Note RPR+ features are only supported when using two GRPs or two PRPs. There is no support for redundancy when a GRP is used together with a PRP.

The default redundancy mode for Cisco 12000 Series Internet Routers is standard RPR. For information on configuring RPR+, refer to the following document:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120limit/120st/120st17/rpr_plus.htm

RPR+ is not supported on all Cisco 12000 Series Internet Router line cards. Refer to the *Release Notes for Cisco IOS Release 12.0S* for the list of supported line cards.

Cisco Nonstop Forwarding

Cisco Nonstop Forwarding (NSF) is a complementary feature to the Stateful Switchover (SSO) feature in Cisco IOS software. NSF always runs together with SSO and works with SSO to minimize the amount of time a network is unavailable to its users following a switchover. The main objective of NSF is to continue forwarding IP packets following a Route Processor (RP) switchover.

Usually, when a networking device restarts, all routing peers of that device detect that the device went down and then came back up. This transition results in what is called a routing flap, which could spread across multiple routing domains. Routing flaps caused by routing restarts create routing instabilities, which are detrimental to the overall network performance. NSF helps to suppress routing flaps in SSO-enabled devices, thus reducing network instability.

NSF allows data packets to continue forwarding along known routes while the routing protocol information is being restored following a switchover. With NSF, peer networking devices do not experience routing flaps. Data traffic is forwarded through intelligent line cards or dual forwarding processors (FPs) while the standby RP assumes control from the failed active RP during a switchover. The ability of line cards and FPs to remain up through a switchover and to be kept current with the Forwarding Information Base (FIB) on the active RP is key to NSF operation.

For information about the SSO feature, refer to this url:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120limit/120s/120s22/sso120s.htm

For more information about Cisco NSF, refer to the Cisco Nonstop Forwarding document at the following location:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120limit/120s/120s22/nsf120s.htm

PCMCIA Flash Memory Tasks

The linear Flash memory card or Flash disk that shipped with your router contains the default Cisco IOS software image you need to boot your router. Flash disks provide higher capacity and better performance than linear Flash memory cards, but otherwise they function almost the same. When using a linear Flash memory card, the Cisco IOS command to identify and access the card is **slot0:** or **slot1:**, depending on the PCMCIA slot in use. When using a Flash disk, those commands are replaced with **disk0:** or **disk1:**.

Note Use only Type I or Type II Flash memory cards.

The following sections describe common software tasks that involve Flash memory cards and Flash disks.

- Formatting a Flash Memory Card or Flash Disk, page 2-11
- Booting from Flash Memory, page 2-12
- Manipulating Files on a Flash Memory Card or Flash Disk, page 2-13

Note For information regarding available Flash memory cards and Flash disks, and how to install them into the RP on your router, refer to the installation and configuration guide available for your Cisco 12000 Series Internet Router.

Formatting a Flash Memory Card or Flash Disk

Before you can use a new Flash memory card or Flash disk, you must format it.

Caution The procedure erases all information on a Flash memory card. To prevent the loss of important data that might be stored on a Flash memory card, proceed carefully. If you want to save the data contained on a Flash memory card, copy the data to a server before you format the card.

To format a new Flash memory card or Flash disk in the RP, follow these steps:

	Command	Purpose

Step 1	`Router# format slot0:`
	or
	`Router# format disk0:`
		Formats a new Flash memory card or Flash disk located in slot 0 on the RP. If the card or disk is located in slot 1, use the **slot1:** or **disk1:** keyword.

Step 2	`All sectors will be erased, proceed? [confirm] y`	Confirms the card or disk format.

Step 3	`Enter volume id (up to 30 characters): MyNewCard`
	`Formatting sector n`
	`Format device slot0 completed`
		Assigns the specified volume ID to the card or disk (MyNewCard in this example). The console displays the "Formatting sector n" line as the card or disk is being formatted. When the count reaches 1, the formatting process is complete.

The new linear Flash memory card or Flash disk is now formatted and ready to use.

For complete command descriptions and configuration information, refer to the *Configuration Fundamentals Command Reference* located at

http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/12cgcr/fun_r/index.htm

or the *Configuration Fundamentals Configuration Guide* located at

http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/12cgcr/fun_c/index.htm.

Booting from Flash Memory

To enable booting from a Cisco IOS software image file located on a PCMCIA Flash memory card or Flash disk, follow these steps:

	Command	Purpose
Step 1	`Router# configure terminal` `Enter configuration commands, one per line. End with CNTL/Z.`	Enters global configuration mode.
Step 2	`Router(config)# boot system flash slot0:`*`filename`*	Specifies that at startup, the router loads the software image named *filename*, located on the linear Flash memory card in slot 0. Use **slot1:** for a Flash memory card in slot 1, **disk0:** for a Flash disk in slot 0, and **disk1:** for a Flash disk in slot 1.
Step 3	`Router(config)# config-register 0x0102`	Disables the Break function and enables the **boot system flash** command.
Step 4	`Router(config)# Ctrl-z`	Exits global configuration mode.

Step 5	`Router# copy running-config startup-config`
	or
	`Router# write memory`
		Saves the software configuration register settings to NVRAM.

Step 6	`Router# reload`	Reboots the router and uses the specified image on the Flash memory card inserted in PCMCIA slot 0 to boot the system.

Note By default, the router boots from a software image file on a Flash memory card or Flash disk located in slot 0.

Manipulating Files on a Flash Memory Card or Flash Disk

You can copy and move files to and from Flash memory cards and Flash disks as you would to any file system. Use any of the following commands to manipulate files on Flash memory cards or Flash disks.

Command	Purpose
`Router# pwd`	Displays the current setting of the **cd** command. Use this command to determine whether the present working directory you are accessing is the onboard Flash SIMM on the RP or a PCMCIA Flash memory device in a slot on the RP. If the router returns *slot0:*, this indicates that you are accessing a PCMCIA linear Flash memory card inserted in slot 0 of the RP.
`Router# cd slot1:`	Changes the present working directory you are accessing to the PCMCIA linear Flash memory card inserted in slot 1 of the RP. Use **disk1:** to access a Flash disk located in slot 1 of the RP.
`Router# cd bootflash:`	Changes the present working directory you are accessing to the Flash memory SIMM on the RP.
`Router# dir`	Displays the directory contents of the Flash memory media in use.
`Router# delete slot0:`*filename*	Deletes the file *filename* from a linear Flash memory card in slot 0. Use **slot1:** for a card in slot 1, **disk0:** for a disk in slot 0, and **disk1:** for a disk in slot 1. Files that are deleted are removed from the directory list, but are not erased permanently. You can use the **undelete** command to recover deleted files.
`Router# squeeze slot0:`	Permanently removes deleted files from a linear Flash memory card located in slot 0. The **squeeze** command also makes all other undeleted files on the Flash card contiguous. Use **slot1:** for a card in slot 1. The squeeze command is not necessary when using Flash disks.

Upgrading and Backing Up Cisco IOS Software Images and Configuration Files

The following sections describe common software tasks that involve upgrading and backing up files on your router:

- Upgrading a Cisco IOS Software Image on Flash Memory from a TFTP Server, page 2-14
- Copying Cisco IOS Software Images Between Flash Memory Cards or Flash Disks, page 2-15
- Specifying a Cisco IOS Software Image File as the Default Boot Image, page 2-16
- Upgrading the Boot Image in SIMM (Bootflash), page 2-17
- Saving a Configuration File, page 2-18
- Restoring a Configuration File, page 2-19
- Recovering from Locked Blocks in Flash Memory Cards or Flash Disks, page 2-19

Upgrading a Cisco IOS Software Image on Flash Memory from a TFTP Server

Your router is shipped with a default Cisco IOS software image preloaded into the Flash memory card or Flash disk located in slot 0 of the RP. As future releases of Cisco IOS software become available, you can download them from Cisco.com to upgrade your router software. You can save these image files to a TFTP server on your network and subsequently download them to the routers on your network.

Note Flash memory cards and Flash disks must be formatted before they are used for the first time. If you have not formatted the Flash memory card or Flash disk, see the "Formatting a Flash Memory Card or Flash Disk" section on page 2-11.

Note To avoid a situation where the router does not have a valid Cisco IOS software image to boot from, make sure that you retain the current software image on the Flash memory card or disk. If you do not have room on the Flash memory card for a second image, save a copy of the image on a separate Flash memory card or Flash disk, or on a TFTP server on your network. See the "Copying Cisco IOS Software Images Between Flash Memory Cards or Flash Disks" section on page 2-15.

To upgrade a Cisco IOS software image file on Flash memory from a TFTP server, follow these steps:

	Command	Purpose
Step 1	`Router# show slot0:`	Verifies that there is sufficient room on the Flash memory card or Flash disk to copy the new software image. Default Flash memory cards are 20M in size. If you see that there is sufficient room for the image that you are copying, continue with the **copy** command in Step 4. Otherwise, continue with the next step.
Step 2	`Router# delete slot0:`*filename* or `Router# delete disk0:`*filename*	For Flash memory cards, (**slot0:** syntax) marks the file *filename* to be deleted; for Flash disks, (**disk0:** syntax) deletes the file *filename*. Delete enough files so that there will be sufficient space on the Flash memory card or Flash disk for the new software image. If you need the files for future use, be sure to copy them to another storage media before you delete them.
Step 3	`Router# squeeze slot0:`	Permanently removes deleted files from a Flash memory card located in slot 0. The **squeeze** command also makes all other undeleted files on the Flash card contiguous. This step is not required if you are using a Flash disk.
Step 4	`Router# copy tftp: slot0:`	Begins the copy dialog to copy a file from a TFTP server to a linear Flash memory card located in slot 0 of the RP. Use **slot1:** to copy to a linear Flash memory card in slot 1, **disk0:** to copy to a Flash disk in slot 0, and **disk1:** to copy to a Flash disk in slot 1.
Step 5	`Address or name of remote host []? 192.168.16.254`	Specifies the address of the TFTP server from which to copy the Cisco ISO software image. The IP address specified here is an example only.
Step 6	`Source filename []? gsr-p-mz.120-7.4.5`	Specifies the name of the file containing the Cisco ISO software image. In this example, the file is named gsr-p-mz.120-7.4.5.

	Command	Purpose
Step 7	`Destination filename [gsr-p-mz.120-7.4.5]?` **`<Return>`**	Specifies the name of the file to be created on the Flash memory media. Press **Return** to use the same name as the source file. After you enter the name of the destination file, the router begins to copy the file. When the privileged EXEC prompt is displayed (`Router#`), the copy is complete.
Step 8	`Router#` **`configure terminal`** `Enter configuration commands, one per line. End with CNTL/Z.`	Enters global configuration mode.
Step 9	`Router(config)#` **`boot system flash slot0:gsr-p-mz.120-7.4.5`**	Specifies that at startup, the router loads the software image named gsr-p-mz.120-7.4.5, located on the Flash memory card in slot 0.
Step 10	`Router(config)#` **`config-register 0x0102`**	Disables the Break function and enables the **boot system flash** command (specified in the previous Step).
Step 11	`Router(config)#` **`Ctrl-Z`**	Exits global configuration mode.
Step 12	`Router#` **`copy running-config startup-config`** `or` `Router#` **`write memory`**	Saves the software configuration register settings to NVRAM.
Step 13	`Router#` **`reload`**	Reboots the router and uses the specified image on the Flash memory card inserted in PCMCIA slot 0 to boot the system.

This completes the Cisco IOS software upgrade procedure.

Note If there is not enough room on the Flash memory card or Flash disk to perform the copy, an error message is displayed to the console. In this case, you will need to delete more files to make room for the new software image, as described in Step 2 and Step 3 of the procedure above.

Copying Cisco IOS Software Images Between Flash Memory Cards or Flash Disks

You can store a Cisco IOS software image on a Flash memory card or Flash disk for backup purposes. You can then copy the software image to the Flash memory card or Flash disk that you are using in your router.

Note Flash memory cards and Flash disks must be formatted before they are used for the first time. If you have not formatted the Flash memory card or Flash disk, see the "Formatting a Flash Memory Card or Flash Disk" section on page 2-11.

Note Copying images between flash memory cards can take a long time.

To copy a Cisco IOS software image file from one Flash memory card or Flash disk to another, follow these steps:

	Command	Purpose
Step 1	`Router# copy slot1: slot0:`	Begins the copy dialog to copy a file on Flash memory card located in slot 1 to the Flash memory card located in slot 0. Use **disk0:** to copy to or from a Flash disk in slot 0, and **disk1:** to copy to or from a Flash disk in slot 1.
Step 2	`Source filename []? gsr-p-mz.120-7.4.5`	Specifies the name of the file containing the Cisco ISO software image on slot 1. In this example, the file is named gsr-p-mz.120-7.4.5.
Step 3	`Destination filename [gsr-p-mz.120-7.4.5]? <Return>`	Specifies the name of the file to be created on the Flash memory card in slot 0. Press **Return** to use the same name as the source file. After you enter the name of the destination file, the router begins to copy the file. This procedure can take a very long time. When the privileged EXEC prompt is displayed (Router#), the copy is complete.

Note If there is not enough room on the Flash memory card or Flash disk to perform the copy, an error message is displayed to the console. In this case, you will need to delete files to make room for the new software image, as described in the "Upgrading a Cisco IOS Software Image on Flash Memory from a TFTP Server" section on page 2-14.

To designate the copied image file to be the new default system image for boot purposes, see the "Specifying a Cisco IOS Software Image File as the Default Boot Image" section on page 2-16.

Specifying a Cisco IOS Software Image File as the Default Boot Image

After you copy a Cisco IOS software image file to a Flash memory card or Flash disk, you will want to designate this new file as the new default system image for boot purposes. To do this, follow these steps:

	Command	Purpose
Step 1	`Router# configure terminal` `Enter configuration commands, one per line. End with CNTL/Z.`	Enters global configuration mode.
Step 2	`Router(config)# boot system flash slot0:gsr-p-mz.ME12_SRP_VER_12_08_12`	Specifies that at startup, the router loads the software image named gsr-p-mz.ME12_SRP_VER_12_08_12, located on the linear Flash memory card in slot 0.
Step 3	`Router(config)# config-register 0x0102`	Disables the Break function and enables the **boot system flash** command.
Step 4	`Router(config)# Ctrl-Z`	Exits global configuration mode.

Step 5	`Router# copy running-config startup-config`
	`or`
	`Router# write memory`
		Saves the software configuration register settings to NVRAM.

Step 6	`Router# reload`	Reboots the router and uses the specified image on the Flash memory card inserted in PCMCIA slot 0 to boot the system.

Upgrading the Boot Image in SIMM (Bootflash)

A mini-Cisco IOS software image is preloaded into the single inline memory module (SIMM) or bootflash. This image can be used to boot the router if there is no other valid image available. It provides a limited number of configuration commands that you can use to locate a valid Cisco IOS software image or correct basic configuration problems. This image provides no routing capabilities.

Caution You must have a valid copy of a boot image in the onboard Flash memory SIMM (bootflash) in order to boot the router. If you delete the current version of the boot image, you must copy in a new image before you reboot the router.

If it becomes necessary to upgrade this image, download a new image from Cisco.com. Use the following procedure to upgrade the boot image in the bootflash with the new image:

Step 1	`Router# delete bootflash:gsr-boot-mz.120-8.S`	Deletes the current bootflash image from the bootflash. In this example, the name of the image is *gsr-boot-mz.120-8.S*. Since there is generally not enough room in the bootflash for more than one image, you will need to delete the current image before you can copy in a new one. If you attempt the copy and there is not enough room in the bootflash for the image, an error message will be displayed.
Step 2	`Router# squeeze bootflash:`	Permanently deletes all files in the bootflash that are marked as deleted.
Step 3	`Router# copy tftp: bootflash:`	Begins the copy dialog to copy a file from a TFTP server to the online Flash memory SIMM (bootflash).
Step 4	`Address or name of remote host []? 192.168.16.254`	Specifies the address of the TFTP server from which to copy the Cisco ISO software image.
Step 5	`Source filename []? gsr-boot-mz.120-21.S`	Specifies the name of the file containing the boot image. In this example, the file is named gsr-boot-mz.120-21.S.bin.
Step 6	`Destination filename [gsr-boot-mz.120-21.S]? <Return>`	Specifies the name of the file to be created on the Flash memory media. Press **Return** to use the same name as the source file. After you enter the name of the destination file, the router begins to copy the file. When the privileged EXEC prompt is displayed (Router#), the copy is complete.
Step 7	`Router# reload`	Reboots the router. The new bootflash image is now ready to be used.

Note Instead of using the **delete** and **squeeze** commands in Step 1 and Step 2, you can use the **format bootflash:** command. Note, however, that the **format** command deletes all crashinfo files located in the bootflash.

This completes the procedure for upgrading the boot image in the onboard Flash memory SIMM.

Saving a Configuration File

It is a good practice to save your configuration file, in case you need to restore it for any reason. You should save your configuration file before you make major changes to the configuration. You have two configuration files: the startup configuration file located in NVRAM and the running configuration file located in DRAM. These will generally be the same, unless you are in the process of changing the configuration.

To save your startup configuration file, use one of the following commands, depending on where you want to save the file:

Command	Purpose
Router# **copy startup-config slot0:***filename*	Copies the configuration file located in NVRAM (the system default) to the PCMCIA Flash memory card in slot 0.
Router# **copy startup-config slot1:***filename*	Copies the configuration file located in NVRAM (the system default) to the PCMCIA Flash memory card in slot 1.
Router# **copy startup-config disk0:***filename*	Copies the configuration file located in NVRAM (the system default) to the PCMCIA Flash memory disk in slot 0.
Router# **copy startup-config disk1:***filename*	Copies the configuration file located in NVRAM (the system default) to the PCMCIA Flash memory disk in slot 1.
Router# **copy startup-config tftp:**	Copies the configuration file located in NVRAM (the system default) to a TFTP server on the network. You will be prompted to provide the address of the TFTP server and the file name.

To save your running configuration file, use one of the following commands, depending on where you want to save the file:

Command	Purpose
Router# **copy running-config slot0:***filename*	Copies the configuration file located in DRAM (the system default) to the PCMCIA Flash memory card in slot 0.
Router# **copy running-config slot1:***filename*	Copies the configuration file located in DRAM (the system default) to the PCMCIA Flash memory card in slot 1.
Router# **copy running-config disk0:***filename*	Copies the configuration file located in DRAM (the system default) to the PCMCIA Flash memory disk in slot 0.
Router# **copy running-config disk1:***filename*	Copies the configuration file located in DRAM (the system default) to the PCMCIA Flash memory disk in slot 1.
Router# **copy running-config tftp:**	Copies the configuration file located in DRAM (the system default) to a TFTP server on the network. You will be prompted to provide the address of the TFTP server and the file name.

Use the **dir** command to verify that the configuration file was copied correctly to the Flash memory media, as shown in the following example for the Flash memory card in slot 0:

```
Router# dir slot0:
-#- -length- -----date/time------ name
1   5200084  May 10 1997 19:24:12 gsr-p-mz.112-8
3   1215     May 10 1997 20:30:52 myfile1
4   6176844  May 10 1997 23:04:10 gsr-p-mz.112-8.1
5   1186     May 10 1997 16:56:50 myfile2

9197156 bytes available (11381148 bytes used)
```

Restoring a Configuration File

To restore a configuration file from a Flash memory card or disk in PCMCIA slot 0 or slot 1 to NVRAM, follow these steps:

	Command	Purpose
Step 1	`Router# copy slot0:`*filename* `startup-config`	Copies the configuration file located in the PCMCIA Flash memory card in slot 0 to NVRAM (the system default). Use **slot1:** to copy from a card in slot 1, **disk0:** to copy from a disk in slot 0, **disk1:** to copy from a disk in slot 1.
Step 2	`Router# copy startup-config running-config`	Designates the startup configuration file stored in NVRAM to be the default running configuration file for the system.

This completes the procedure for restoring a configuration file from a Flash memory card or disk to NVRAM.

Recovering from Locked Blocks in Flash Memory Cards or Flash Disks

A locked block in a Flash memory card or Flash disk occurs when power is lost or a Flash memory card or Flash disk is removed from its PCMCIA slot on the RP during a write or erase operation.

When a block of Flash memory is locked, it cannot be written to or erased. Any attempt at such an operation will consistently fail at the blocked location. The only way to recover from locked blocks in a Flash memory card or Flash disk is to reformat it using the **format** command. For more information, see the "Formatting a Flash Memory Card or Flash Disk" section on page 2-11.

Caution Formatting a Flash memory card or disk erases all existing data on the card or disk.